Lab1

Zara Waheed

4th Feb 2022

```
summary(cars)
```

```
##      speed           dist       
##  Min.   : 4.0   Min.   :  2.00  
##  1st Qu.:12.0   1st Qu.: 26.00  
##  Median :15.0   Median : 36.00  
##  Mean   :15.4   Mean   : 42.98  
##  3rd Qu.:19.0   3rd Qu.: 56.00  
##  Max.   :25.0   Max.   :120.00  
```

Question 1:

The intercept represents the number of sales if all mother predictors are at 0, which from the p-value we can gauge that is very unlikely to happen in real life. TV is likely to increase in 46 sales units per $1000 spent on advertising. Every $1000 spent on radio advertising increases sales by around 189 units.Spending no money on Radio or TV advertising is very unlikely as can be seen from the p values. Newspaper advertising is meant to have a negative effect by 1 sale unit on sales but the p value shows us that there is a high chnace that the relationship between sales and newspaper advertising is not significant.

Question 2:

KNN classification tries to predict the class to which the output variable belongs to by finding out the nearest points probability. KNN regression tries to predict the value of the output variable by using an nearest points average.

Question 5

We add both equations together by substituting beta into the `y^_i` equation. We get `x_i (sum x_j y_j)/( sum x_k^2)`. `x_i` is added to the summation. Now `a_j` is set equal to `(x_i x_j)/sum x_k^2` and we get `y^_i = sum a_j y_j`

Question 6

If x axis is shifted by `mean(x)`, the y axis should equal `mean(y)` given `B_1`. To show this let's plug in `f(mean(x))` into the linear equation which gives us `x=mean(x) y=B_0 + B_1 mean(x)`. Plugging the optimal values of `B_0` into the equation gives us `y=(mean(y) - B_1*mean(x)) + B_1*mean(x)`. That gives us `y=mean(y)` when `x=mean(x)`.

Question 11

```
set.seed(100)
x=rnorm(100)
y=2*x+rnorm(100)
```

a)

```
fit11a=lm(y~x+0)
summary(fit11a)
```

```
##
## Call:
## lm(formula = y ~ x + 0)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.04051 -0.42120 -0.06707  0.49725  1.95009
##
## Coefficients:
##   Estimate Std. Error t value Pr(>|t|)
## x  1.89466    0.07769   24.39   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.789 on 99 degrees of freedom
## Multiple R-squared:  0.8573, Adjusted R-squared:  0.8559
## F-statistic: 594.8 on 1 and 99 DF,  p-value: < 2.2e-16
```

The estimate is 1.89 when we 2x gives us y. This shows a good fit because the probability that we actually did not multiply x by anything as in the p-value, is very low and the t-statistic is high.

b)

```
fit11b=lm(x~y+0)
summary(fit11b)
```

```
##
## Call:
## lm(formula = x ~ y + 0)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.17839 -0.22598  0.01977  0.21129  1.10008
##
## Coefficients:
##   Estimate Std. Error t value Pr(>|t|)
## y  0.45249    0.01855   24.39   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3856 on 99 degrees of freedom
## Multiple R-squared:  0.8573, Adjusted R-squared:  0.8559
## F-statistic: 594.8 on 1 and 99 DF,  p-value: < 2.2e-16
```

The coefficient is much smaller than the previous case because of the switching of the y and x. This also shows a good fit but a bit less than the previous one. However, the probability that we actually did not multiply x by anything as in the p-value, is very low and the t-statistic is high.

c)

These two results are almost inverses of each other.

d)

B^=(sum_j x_j y_j)/(sum_k x^2_k)

y_i^2 + 2x_i*B^*y_i+x_iB^^2

...

e)

I we substitute x for y in the equation gives us the exact same equation. Therefore, the t-statistic would also be the same for both cases.

f)

```
fit11f.1=lm(y~x)
fit11f.2=lm(x~y)
t1=summary(fit11f.1)$coefficients[2,3]
t2=summary(fit11f.2)$coefficients[2,3]
t1
```

```
## [1] 24.2674
```

```
t2
```

```
## [1] 24.2674
```

Question 12

a)

If the coefficient is 1, they would be the same.

b)

```
x=rnorm(100)
y=0.5*x+rnorm(100)
fit12b.1 <- lm(x~y+0)
fit12b.2 <- lm(y~x+0)
summary(fit12b.1)
```

```
##
## Call:
## lm(formula = x ~ y + 0)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.21839 -0.65809  0.03318  0.74361  2.94409
##
## Coefficients:
##   Estimate Std. Error t value Pr(>|t|)
## y  0.31316    0.08538   3.668 0.000396 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9752 on 99 degrees of freedom
## Multiple R-squared:  0.1196, Adjusted R-squared:  0.1108
## F-statistic: 13.45 on 1 and 99 DF,  p-value: 0.0003957
```

```
summary(fit12b.2)
```

```
##
## Call:
## lm(formula = y ~ x + 0)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.7339 -0.6612 -0.1015  0.6909  3.3539
##
## Coefficients:
##   Estimate Std. Error t value Pr(>|t|)
## x   0.3821     0.1042   3.668 0.000396 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.077 on 99 degrees of freedom
## Multiple R-squared:  0.1196, Adjusted R-squared:  0.1108
## F-statistic: 13.45 on 1 and 99 DF,  p-value: 0.0003957
```

c)

```
x=rnorm(100)
y=1*x
fit12c.1 <- lm(x~y+0)
fit12c.2 <- lm(y~x+0)
summary(fit12c.1)
```

```
## Warning in summary.lm(fit12c.1): essentially perfect fit: summary may be
## unreliable
```

```
##
## Call:
## lm(formula = x ~ y + 0)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -3.546e-16 -4.819e-17 -2.070e-18  3.712e-17  5.345e-16
##
## Coefficients:
##   Estimate Std. Error   t value Pr(>|t|)
## y 1.00e+00   9.45e-18 1.058e+17   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.015e-16 on 99 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 1.12e+34 on 1 and 99 DF,  p-value: < 2.2e-16
```

```r
summary(fit12c.2)
```

```
## Warning in summary.lm(fit12c.2): essentially perfect fit: summary may be
## unreliable

##
## Call:
## lm(formula = y ~ x + 0)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -3.546e-16 -4.819e-17 -2.070e-18  3.712e-17  5.345e-16
##
## Coefficients:
##   Estimate Std. Error   t value Pr(>|t|)
## x 1.00e+00   9.45e-18 1.058e+17   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.015e-16 on 99 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 1.12e+34 on 1 and 99 DF,  p-value: < 2.2e-16
```

Question 13

a)

```r
x=rnorm(100)
```

b)

```r
eps=rnorm(100,0,0.25)
```

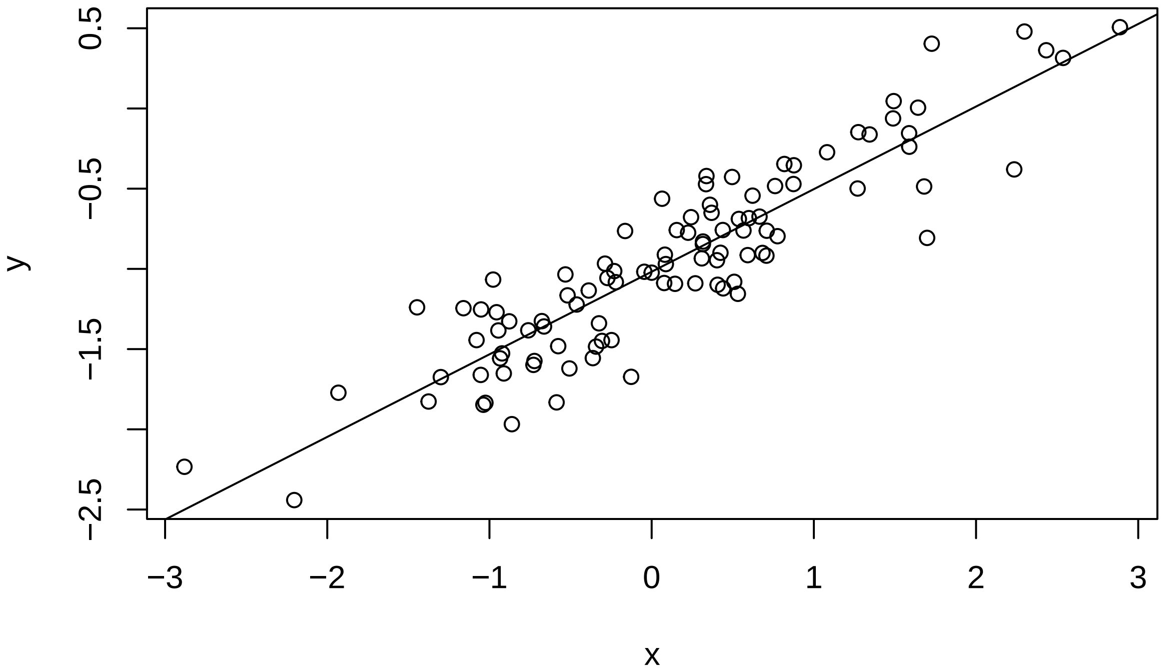
c)

```r
y=-1+0.5*x+eps
```

y = 100 B0 = -1 B1 = 0.5

d)

```r
plot(x,y)
abline(lm(y~x))
```



e)

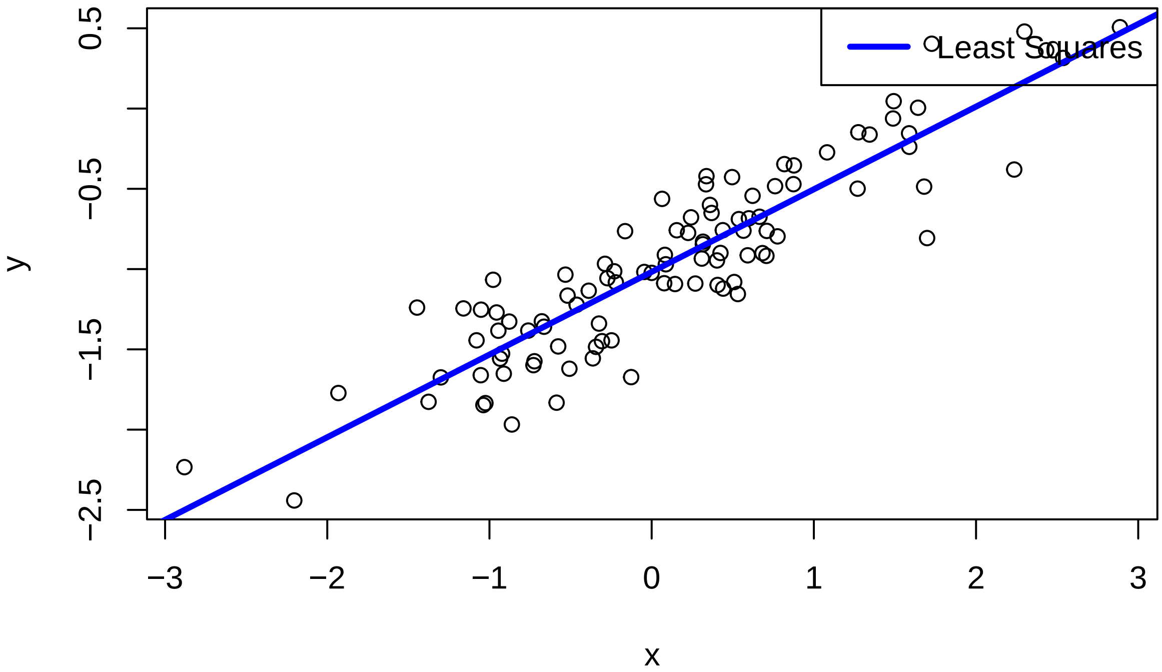
```
fit13e=lm(y~x)
summary(fit13e)
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.66345 -0.17002  0.02593  0.17785  0.53243
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.01762    0.02546  -39.97   <2e-16 ***
## x            0.51482    0.02436   21.13   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2528 on 98 degrees of freedom
## Multiple R-squared:   0.82,  Adjusted R-squared:  0.8182
## F-statistic: 446.6 on 1 and 98 DF,  p-value: < 2.2e-16
```

The predicted B0 and B1 are very close to the actual B0 and B1.

f)

```
plot(x,y)
abline(lm(y~x),col="blue",lwd=3)
legend("topright", legend="Least Squares", lty=1, lwd=3, col="blue")
```



g)

```
fit13g=lm(y~poly(x,2))
summary(fit13g)
```

```
##
## Call:
## lm(formula = y ~ poly(x, 2))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.68031 -0.16673  0.03116  0.17114  0.51428
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.95376    0.02529 -37.715   <2e-16 ***
## poly(x, 2)1  5.34207    0.25288  21.125   <2e-16 ***
## poly(x, 2)2  0.24403    0.25288   0.965    0.337
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2529 on 97 degrees of freedom
## Multiple R-squared:  0.8218, Adjusted R-squared:  0.8181
## F-statistic: 223.6 on 2 and 97 DF,  p-value: < 2.2e-16
```

The adjusted `R^2` value is worse and the high p-value of `x^2` shows that the fit is not great.

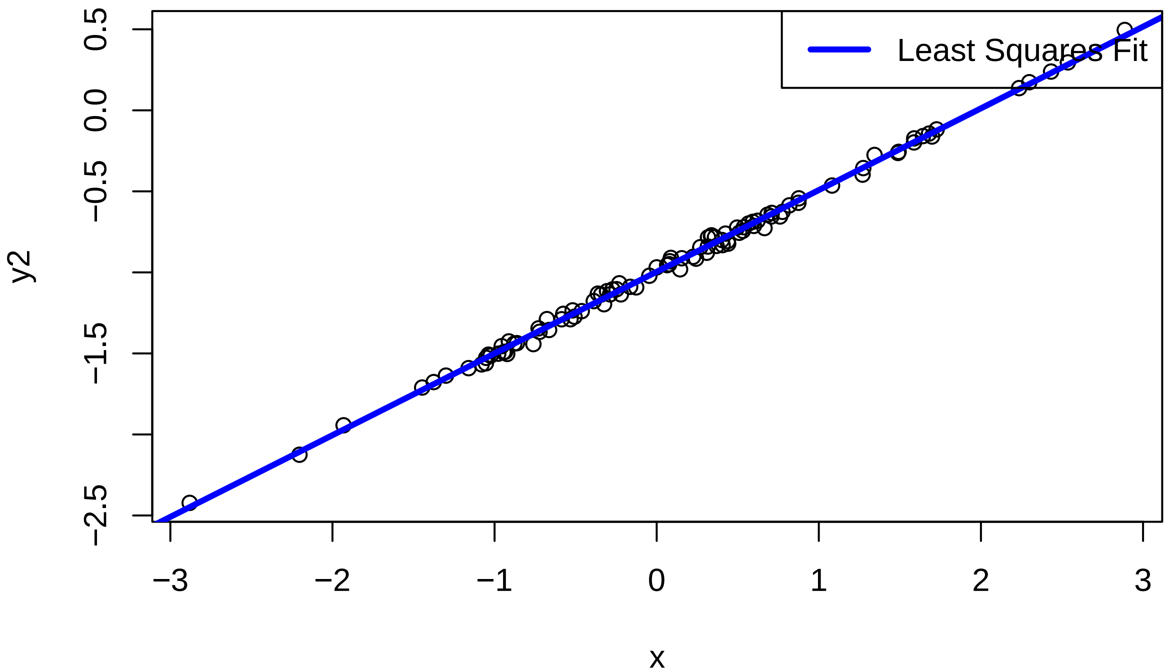
h)

```
eps2=rnorm(100,0,0.025)
y2=-1+0.5*x+eps2
fit13h=lm(y2~x)
```

```
summary(fit13h)
```

```
##
## Call:
## lm(formula = y2 ~ x)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.065489 -0.019323  0.001845  0.017429  0.054577
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.996293   0.002727  -365.3   <2e-16 ***
## x            0.504156   0.002610   193.2   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.02708 on 98 degrees of freedom
## Multiple R-squared:  0.9974, Adjusted R-squared:  0.9974
## F-statistic: 3.732e+04 on 1 and 98 DF,  p-value: < 2.2e-16
```

```
plot(x,y2)
abline(fit13h,col="blue",lwd=3)
legend("topright", legend="Least Squares Fit", lty=1, lwd=3, col="blue")
```



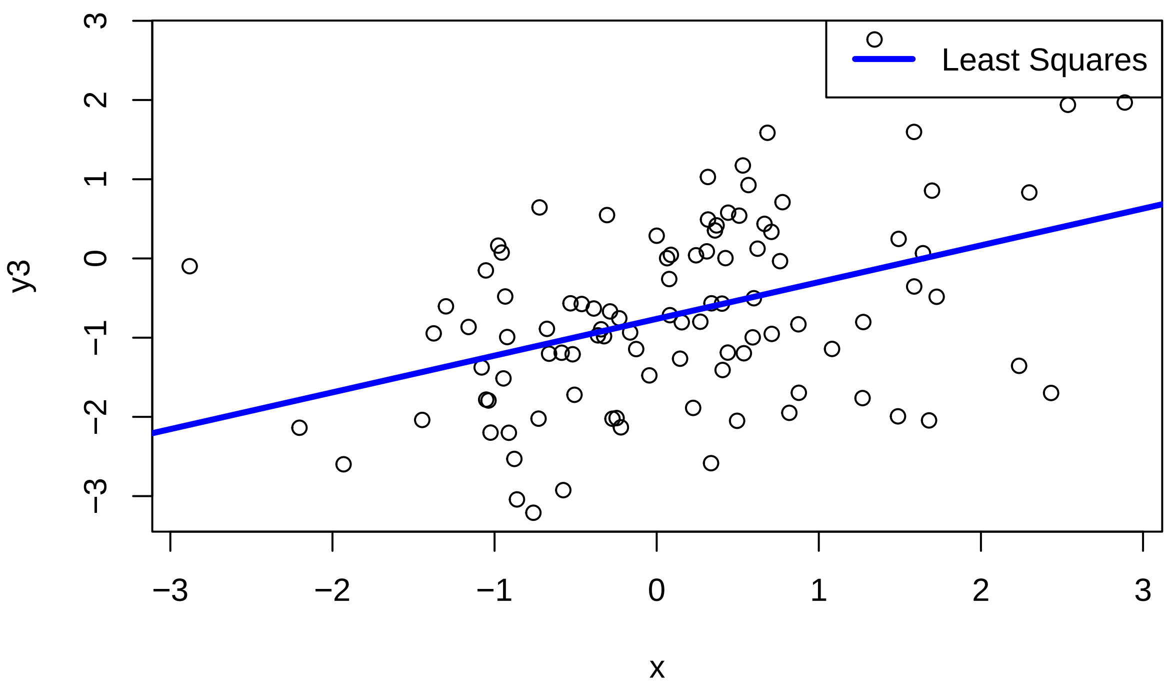
The estimate of the previous model was better. Also the points are very close to the best fit least-squares line.

i)

```
eps3=rnorm(100,0,1)
y3=-1+0.5*x+eps3
fit13i=lm(y3~x)
summary(fit13i)
```

```
##
## Call:
## lm(formula = y3 ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.09438 -0.68594 -0.00604  0.76274  2.90336
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.7626     0.1085  -7.028 2.81e-10 ***
## x             0.4639     0.1038   4.468 2.12e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.077 on 98 degrees of freedom
## Multiple R-squared:  0.1692, Adjusted R-squared:  0.1607
## F-statistic: 19.96 on 1 and 98 DF,  p-value: 2.122e-05
```

```
plot(x,y3)
abline(fit13i,col="blue",lwd=3)
legend("topright", legend="Least Squares", lty=1, lwd=3, col="blue")
```



The noise makes it hard for the model to estimate the intercept and slope and the estimates are further away from part e) and h). This can be seen in the figure around the best fit line as well.

j)

```
confint(fit13e)
```

```
##                  2.5 %     97.5 %
## (Intercept) -1.0681468 -0.9671001
```

```
## x             0.4664742  0.5631653
```

```
confint(fit13h)
```

```
##                  2.5 %     97.5 %
## (Intercept) -1.0017050 -0.9908810
## x            0.4989772  0.5093347
```

```
confint(fit13i)
```

```
##                  2.5 %     97.5 %
## (Intercept) -0.9778961 -0.5472506
## x            0.2578323  0.6699149
```

The confidence interval increases and decreases with the error.

Question 14

a)

```
set.seed(1)
x1=runif(100)
x2=0.5*x1+rnorm(100)/10
y=2+2*x1+0.3*x2+rnorm(100)
```

The equation of the line is `y=b0+b1*x1+b2*x2+e`. The regression coefficients are 2, 2 and 0.3 for the intercept, x1, and x2, respectively.

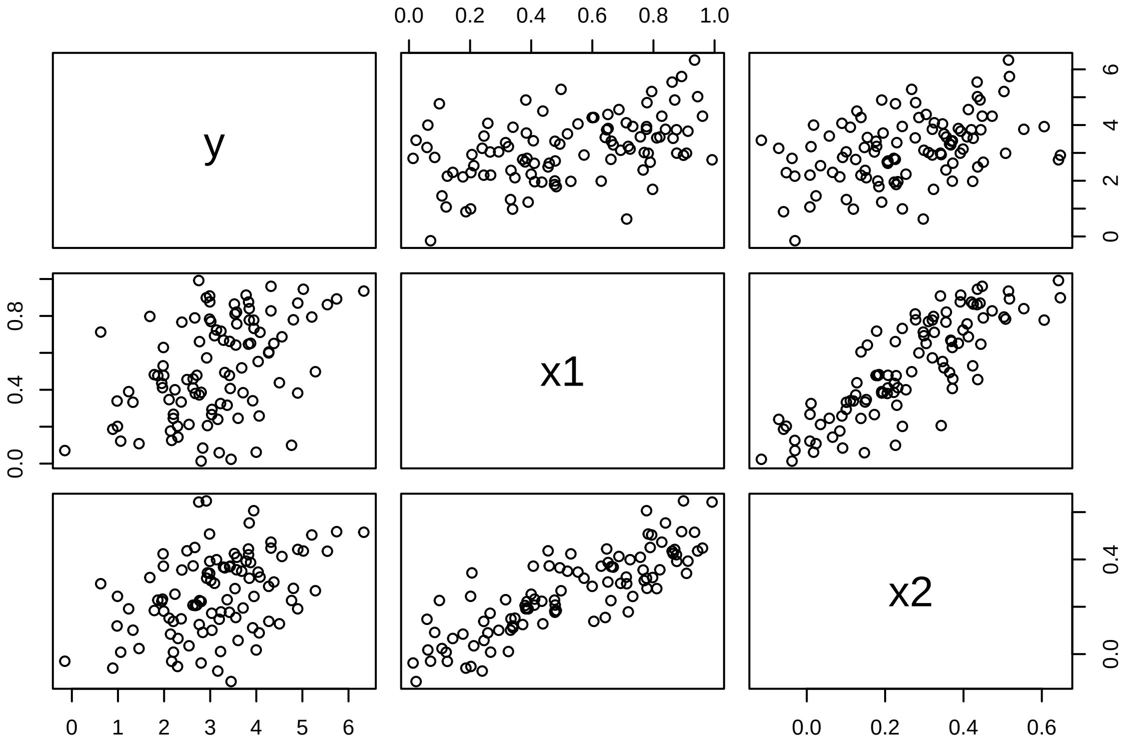
b)

What is the correlation between x1 and x2? Create a scatterplot displaying the relationship between the variables.

```
cor(data.frame(y=y, x1=x1, x2=x2))
```

```
##            y        x1        x2
## y  1.0000000 0.4498446 0.4199171
## x1 0.4498446 1.0000000 0.8351212
## x2 0.4199171 0.8351212 1.0000000
```

```
pairs(data.frame(y=y, x1=x1, x2=x2))
```



x1 and x2 are highly correlated, with a pearson's correlation of 0.8. This colinearity is visable in the pairs plot.

c)

```
fit14c=lm(y~x1+x2)
summary(fit14c)
```

```
##
## Call:
## lm(formula = y ~ x1 + x2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.8311 -0.7273 -0.0537  0.6338  2.3359
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.1305     0.2319   9.188 7.61e-15 ***
## x1            1.4396     0.7212   1.996   0.0487 *
## x2            1.0097     1.1337   0.891   0.3754
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.056 on 97 degrees of freedom
## Multiple R-squared:  0.2088, Adjusted R-squared:  0.1925
## F-statistic:  12.8 on 2 and 97 DF,  p-value: 1.164e-05
```

The intercept is about the same and the null can be rejected based on that. The b1 coefficient is smaller, so rejecting the null is a bit difficult. With b1 it's greater so we cannot reject the null.

d)

```r
fit14d=lm(y~x1)
summary(fit14d)
```

```
##
## Call:
## lm(formula = y ~ x1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.89495 -0.66874 -0.07785  0.59221  2.45560
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.1124     0.2307   9.155 8.27e-15 ***
## x1            1.9759     0.3963   4.986 2.66e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.055 on 98 degrees of freedom
## Multiple R-squared:  0.2024, Adjusted R-squared:  0.1942
## F-statistic: 24.86 on 1 and 98 DF,  p-value: 2.661e-06
```

The intercept is a bit larger, and b1 is a bit smaller, but the adjusted `R2` is better and the `b1` estimate seems more significant. Overall the fit seems better.

e)

```r
fit14e=lm(y~x2)
summary(fit14e)
```

```
##
## Call:
## lm(formula = y ~ x2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.62687 -0.75156 -0.03598  0.72383  2.44890
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.3899     0.1949   12.26  < 2e-16 ***
## x2            2.8996     0.6330    4.58 1.37e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.072 on 98 degrees of freedom
## Multiple R-squared:  0.1763, Adjusted R-squared:  0.1679
## F-statistic: 20.98 on 1 and 98 DF,  p-value: 1.366e-05
```

This one is not as good a fit, but b0 has a low p-value.

f)

Since x1 and x2 are correlated, they are giving a good relationship with y individially but not when they are all meshed together in one equation. The results from c) to e) correspond to that.

g)

```r
x1=c(x1, 0.1)
x2=c(x2, 0.8)
y=c(y,6)
```

```r
fit14g.c=lm(y~x1+x2)
summary(fit14g.c)
```

```
##
## Call:
## lm(formula = y ~ x1 + x2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.73348 -0.69318 -0.05263  0.66385  2.30619
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.2267     0.2314   9.624 7.91e-16 ***
## x1            0.5394     0.5922   0.911  0.36458
## x2            2.5146     0.8977   2.801  0.00614 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.075 on 98 degrees of freedom
## Multiple R-squared:  0.2188, Adjusted R-squared:  0.2029
## F-statistic: 13.72 on 2 and 98 DF,  p-value: 5.564e-06
```

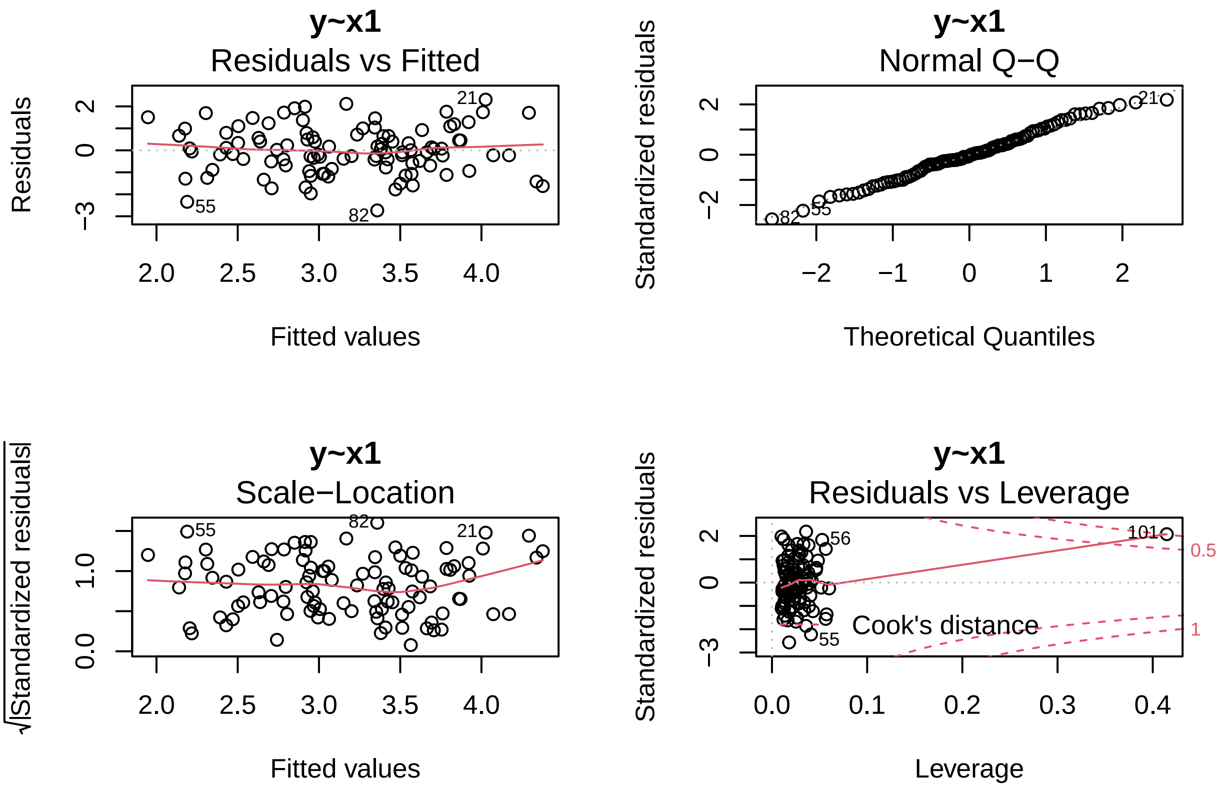
```r
fit14g.d=lm(y~x1)
summary(fit14g.d)
```

```
##
## Call:
## lm(formula = y ~ x1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.8897 -0.6556 -0.0909  0.5682  3.5665
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.2569     0.2390   9.445 1.78e-15 ***
## x1            1.7657     0.4124   4.282 4.29e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.111 on 99 degrees of freedom
## Multiple R-squared:  0.1562, Adjusted R-squared:  0.1477
## F-statistic: 18.33 on 1 and 99 DF,  p-value: 4.295e-05
```

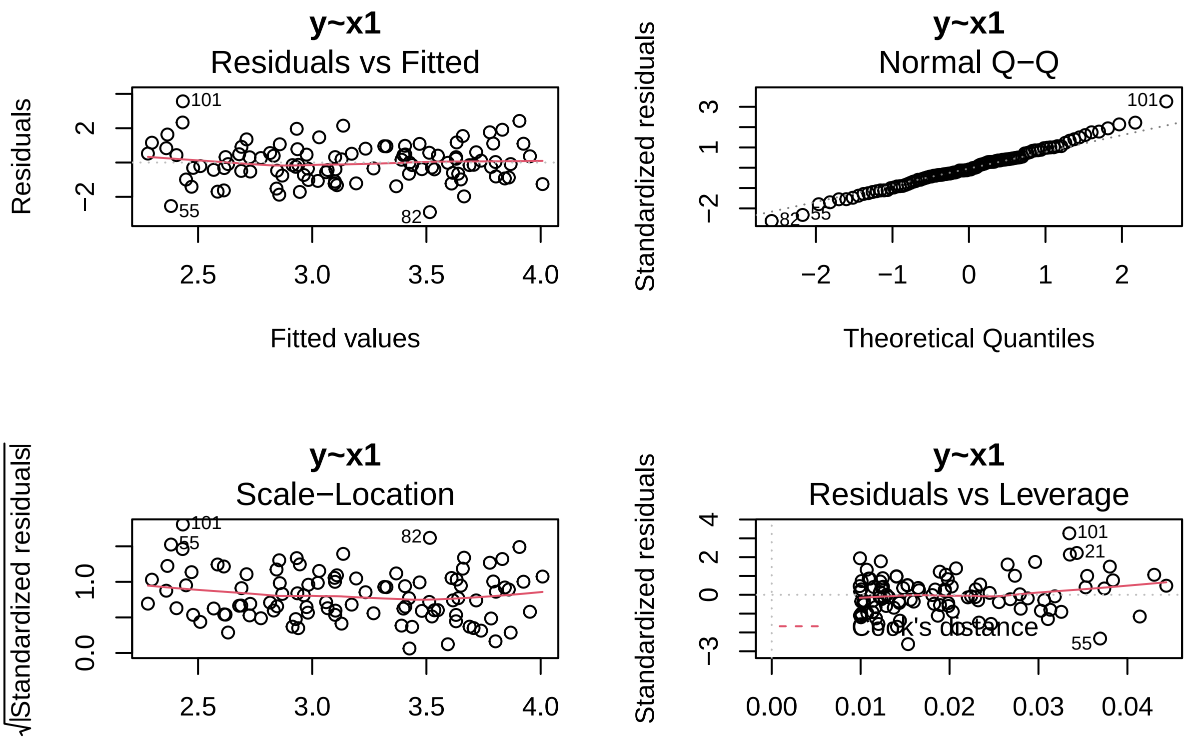
```r
fit14g.e=lm(y~x2)
summary(fit14g.e)
```

```
##
## Call:
## lm(formula = y ~ x2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.64729 -0.71021 -0.06899  0.72699  2.38074
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.3451     0.1912  12.264  < 2e-16 ***
## x2            3.1190     0.6040   5.164 1.25e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.074 on 99 degrees of freedom
## Multiple R-squared:  0.2122, Adjusted R-squared:  0.2042
## F-statistic: 26.66 on 1 and 99 DF,  p-value: 1.253e-06
```

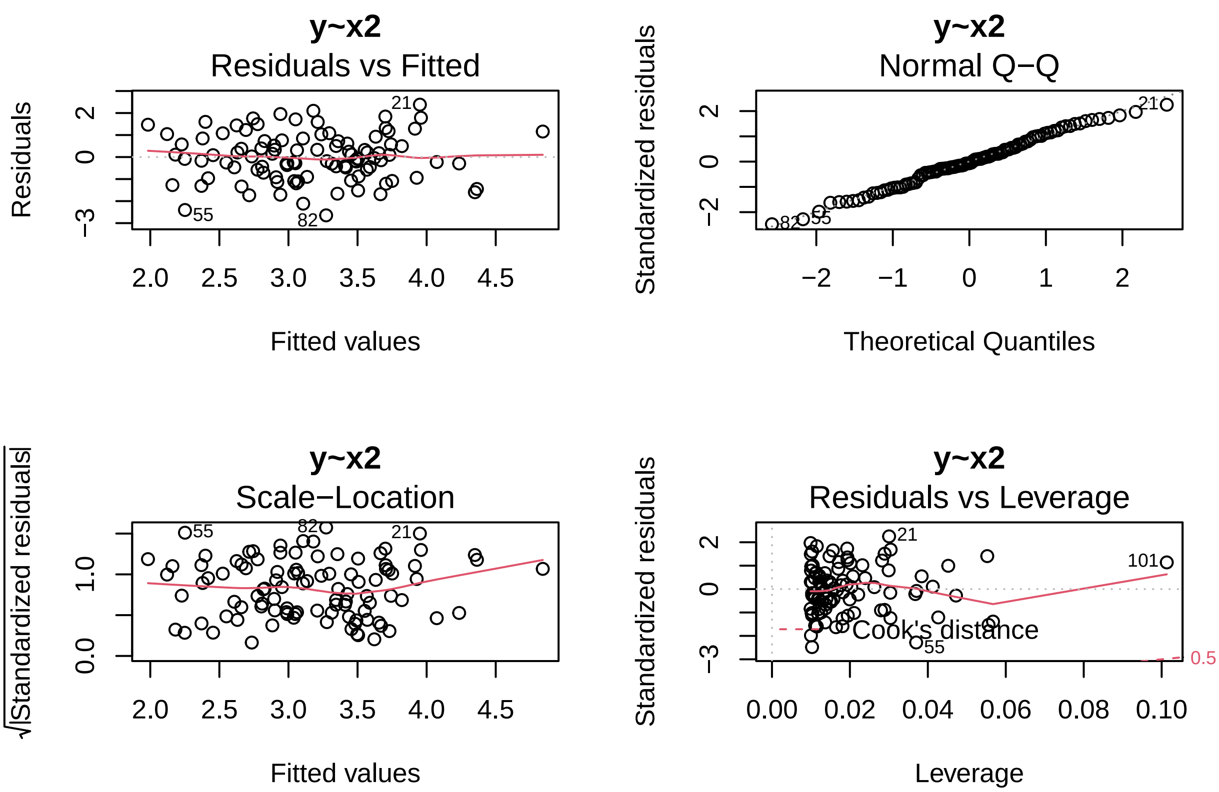
```r
par(mfrow=c(2,2))
plot(fit14g.c, main="y~x1")
```




```r
par(mfrow=c(2,2))
plot(fit14g.d, main="y~x1")
```




```r
par(mfrow=c(2,2))
plot(fit14g.e, main="y~x2")
```



When we include both x1 and x2 in the model, the point does not appear to be an outlier but does have a significant leverage point on the residuals vs leverage plot.

When we have only x1 in the model, the point does appear to be an outlier but no leverage point.

With we have only x2 in the model, the point does not seem to be an outlier, but has a little more leverage.